An optimal class of eighth-order iterative methods based on Kung and Traub's method with its dynamics

Gunar Matthies [a*] Mehdi Salimi [b†] Somayeh Sharifi [c‡]
Juan Luis Varona [d§]

[a]Institut für Numerische Mathematik, Technische Universität Dresden, Germany
[b]Center for Dynamics, Department of Mathematics, Technische Universität Dresden, Germany
[c]Young Researchers and Elite Club, Hamedan Branch, Islamic Azad University, Hamedan, Iran
[d]Departamento de Matemáticas y Computación, Universidad de La Rioja, Logroño, Spain

Abstract

In this paper, we present a three-point without memory iterative method based on Kung and Traub's method for solving non-linear equations in one variable. The proposed method has eighth-order convergence and costs only four function evaluations each iteration which supports the Kung-Traub conjecture on the optimal order of convergence. Consequently, this method possesses very high computational efficiency. We present the construction, the convergence analysis, and the numerical implementation of the method. Furthermore, comparisons with some other existing optimal eighth-order methods concerning accuracy and basins of attraction for several test problems will be given.

Keywords: Multi-point iterative methods; Simple root; Order of convergence; Kung and Traub's conjecture; Basins of attraction.

Mathematics Subject Classification: 65H05, 37F10

1 Introduction

Solving nonlinear equations is a basic and extremely valuable tool in all fields in science and engineering. One can distinguish between two general approaches for solving nonlinear equations numerically, namely, one-step and multi-step methods. Multi-step methods overcome some computational issues encountered with one-step iterative methods. Typically they allow us to achieve a greater accuracy with the same number of function evaluations. Important aspects related to these methods are order of convergence and optimality. Therefore, it is favorable to attain with fixed number of function evaluations each iteration step a convergence order which is as high as possible. A central role in this context plays the unproved conjecture by Kung and Traub [14]. It states that an optimal multi-step method without memory which uses $k+1$ evaluations could achieve a convergence order of 2^k. Considering this conjecture, many optimal two-step and three-step methods have been presented.

In the recent years, a large number of multi-step methods for finding simple roots x^* of a nonlinear equation $f(x) = 0$ with a scalar function $f : D \subset \mathbb{R} \to \mathbb{R}$ which is defined on an open interval D (or $f : D \subset \mathbb{C} \to \mathbb{C}$ defined on a region D in the complex plane $\mathbb{C}$) have been developed and analyzed for improving the convergence order of classical methods. The Newton-Raphson iteration $x_{n+1} := x_n - \frac{f(x_n)}{f'(x_n)}$ is probably the most widely used algorithm for finding roots. It is of second order and

*gunar.matthies@tu-dresden.de
†mehdi.salimi@tu-dresden.de
‡s.sharifi@iauh.ac.ir
§jvarona@unirioja.es

requires two evaluations for each iteration step, one evaluation of f and one of f'. The Newton-Raphson iteration is an example of a one-point iteration, i.e., in each iteration step the evaluations are taken at a single point. The basic optimality theorem for one-point iterations (see Traub [24, § 5.4] or an improved proof in [14]) shows that an analytic one-point iteration based on k evaluations is of order at most k. Thus, the Newton-Raphson iteration is an optimal one-point method with $k = 2$.

Some well known two-point methods without memory are described e.g. in Jarratt [12], King [13], and Ostrowski [18]. Using inverse interpolation, Kung and Traub [14] constructed two general optimal classes without memory. Since then, there have been many attempts to construct optimal multi-point methods, utilizing e.g. weight functions, see in particular [2, 3, 5, 15, 19, 20, 21, 22, 26]. Here, we will construct a class of eighth-order methods free from second order derivatives with efficiency index $\sqrt[4]{8} \simeq 1.68179$; recall that the efficiency index of an iterative method of order p requiring k function evaluations per iteration step is defined by $E(k,p) = \sqrt[k]{p}$, see [18].

The paper is organized as follows: Section 2 is devoted to introduce the ideas for the construction of the new optimal class of eighth-order methods based on Kung and Traub's method by using a Newton-step and suitable weight functions. In Section 3 we give the details of the new methods and investigate the convergence order; this allows to present a class of optimal three-point methods by using suitable weight functions. Particularizing the weight functions we construct a three-parametric family of eighth-order optimal iterative root-finding methods. By assigning particular values to these parameters we propose two examples for this kind of methods. Numerical performance and comparisons with other methods are illustrated in Section 4. In Section 5 we approximate and visualize the basins of attraction of the proposed method and compare them with several existing methods, both graphically and by mean of some numerical measures. Finally, a conclusion is provided in Section 6.

2 Description of the method

In this section we construct a new optimal three-point class of iterative methods for solving nonlinear equations based on Kung and Traub's method [14].

The Kung and Traub's method is given by

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] x_{n+1} := y_n - \dfrac{f(x_n)f(y_n)}{(f(x_n)-f(y_n))^2} \cdot \dfrac{f(x_n)}{f'(x_n)}. \end{cases} \tag{2.1}$$

In (2.1) and all forthcoming methods, the iteration rule is used for $n = 0, 1, \ldots$ and x_0 denotes an initial approximation of the simple root x^*. The convergence order of (2.1) is four with three function evaluations in each iteration step. Hence, this method is optimal. We intend to increase the order of convergence and extend (2.1) by mean of an additional Newton step

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(x_n)f(y_n)}{(f(x_n)-f(y_n))^2} \cdot \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] x_{n+1} := z_n - \dfrac{f(z_n)}{f'(z_n)}. \end{cases} \tag{2.2}$$

Method (2.2) uses five function evaluations with convergence order eight. Consequently, this method is not optimal. In order to decrease the number of function evaluations, we are going to approximate

$f'(z_n)$ by an expression based on $f(x_n)$, $f(y_n)$, $f(z_n)$, and $f'(x_n)$, namely

$$f'(z_n) \approx \frac{f'(x_n)}{J(t_n, u_n)G(s_n)},$$

with $t_n := \frac{f(y_n)}{f(x_n)}$, $u_n := \frac{f(z_n)}{f(x_n)}$, $s_n := \frac{f(z_n)}{f(y_n)}$, and suitable functions J and G.

Therefore, we have

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ \\ z_n := y_n - \dfrac{f(x_n)f(y_n)}{(f(x_n) - f(y_n))^2} \cdot \dfrac{f(x_n)}{f'(x_n)}, \\ \\ x_{n+1} := z_n - \dfrac{f(z_n)}{f'(x_n)} J(t_n, u_n)G(s_n), \end{cases} \tag{2.3}$$

as iteration rule.

3 Convergence analysis

In the following theorem, we analyze the convergence order of method (2.3). In particular, we find the requisites to the weight functions J and G in (2.3) guarantee the requested order eight. Although we enunciate the method for real functions and a real root, the same can be written (with an identical proof) if we have a complex function $f : D \subset \mathbb{C} \to \mathbb{C}$ with a complex root $x^* \in D$.

Theorem 1. *Let $f : D \subset \mathbb{R} \to \mathbb{R}$ be an eight times continuously differentiable function with a simple zero $x^* \in D$, and let $J : \mathbb{R}^2 \to \mathbb{R}$ and $G : \mathbb{R} \to \mathbb{R}$ sufficiently differentiable functions in a neighborhood of the origin. If the initial point x_0 is sufficiently close to x^*. Then, the method defined by (2.3) converges to x^* with order eight if the conditions*

$$J_{0,0} = 1, \quad J_{1,0} = 2, \quad J_{2,0} = 8, \quad J_{0,1} = 2, \quad J_{3,0} = 36,$$

and

$$G_0 = 1, \quad G_1 = 1,$$

with $J_{i,j} = \frac{\partial^{i+j} J(t,u)}{\partial t^i \partial u^j}|_{(t,u)=(0,0)}$ and $G_i = \frac{d^i G(s)}{ds^i}|_{s=0}$ are fulfilled.

Proof. Let $e_n := x_n - x^*$, $e_{n,y} := y_n - x^*$, $e_{n,z} := z_n - x^*$ and $c_n := \frac{f^{(n)}(x^*)}{n! f'(x^*)}$ for $n \in \mathbb{N}$. Using the fact that $f(x^*) = 0$, the Taylor expansion of f at x^* yields

$$f(x_n) = f'(x^*) \left(e_n + c_2 e_n^2 + c_3 e_n^3 + \cdots + c_8 e_n^8\right) + O(e_n^9) \tag{3.1}$$

and

$$f'(x_n) = f'(x^*) \left(1 + 2c_2 e_n + 3c_3 e_n^2 + 4c_4 e_n^3 + \cdots + 9c_9 e_n^8\right) + O(e_n^9). \tag{3.2}$$

Therefore, we have

$$\begin{aligned} \frac{f(x_n)}{f'(x_n)} = e_n &- c_2 e_n^2 + \left(2c_2^2 - 2c_3\right) e_n^3 + (-4c_2^3 + 7c_2 c_3 - 3c_4) e_n^4 \\ &+ (8c_2^4 - 20c_2^2 c_3 + 6c_3^2 + 10c_2 c_4 - 4c_5) e_n^5 \\ &+ (-16c_2^5 + 52c_2^3 c_3 - 28c_2^2 c_4 + 17c_3 c_4 - c_2(33c_3^2 - 13c_5)) e_n^6 + O(e_n^7), \end{aligned}$$

and

$$\begin{aligned} e_{n,y} = y_n - x^* = c_2 e_n^2 &+ (-2c_2^2 + 2c_3) e_n^3 + (4c_2^3 - 7c_2 c_3 + 3c_4) e_n^4 \\ &+ (-8c_2^4 + 20c_2^2 c_3 - 6c_3^2 - 10c_2 c_4 + 4c_5) e_n^5 \\ &+ (16c_2^5 - 52c_2^3 c_3 + 28c_2^2 c_4 - 17c_3 c_4 + c_2(33c_3^2 - 13c_5)) e_n^6 + O(e_n^7). \end{aligned}$$

We have for $f(y_n)$ also

$$f(y_n) = f'(x^*)\left(e_{n,y} + c_2 e_{n,y}^2 + c_3 e_{n,y}^3 + \cdots + c_8 e_{n,y}^8\right) + O(e_{n,y}^9). \tag{3.3}$$

Therefore, by substituting (3.1), (3.2), and (3.3) into (2.2), we get

$$\begin{aligned} e_{n,z} = z_n - x^* = {} & (2c_2^3 - c_2c_3)e_n^4 - 2(5c_2^4 - 7c_2^2c_3 + c_3^2 + c_2c_4)e_n^5 \\ & + \left(31c_2^5 - 726c_2^3c_3 + 21c_2^2c_4 - 7c_3c_4 + c_2(30c_3^2 - 3c_5)\right)e_n^6 + O(e_n^7). \end{aligned}$$

We get for $f(z_n)$ also

$$f(z_n) = f'(x^*)\left(e_{n,z} + c_2 e_{n,z}^2 + c_3 e_{n,z}^3 + \cdots + c_8 e_{n,z}^8\right) + O(e_{n,z}^9). \tag{3.4}$$

From (3.1) and (3.3), we have

$$\begin{aligned} t_n = \frac{f(y_n)}{f(x_n)} = {} & c_2 e_n + (-3c_2^2 + 2c_3)e_n^2 + (8c_2^3 - 10c_2c_3 + 3c_4)e_n^3 \\ & + (-20c_2^4 + 37c_2^2c_3 - 8c_3^2 - 14c_2c_4 + 4c_5)e_n^4 \\ & + \left(48c_2^5 - 118c_2^3c_3 + 51c_2^2c_4 - 22c_3c_4 + c_2(55c_3^2 - 18c_5)\right)e_n^5 + O(e_n^6), \end{aligned} \tag{3.5}$$

and from (3.1) and (3.4), we obtain

$$\begin{aligned} u_n = \frac{f(z_n)}{f(x_n)} = {} & (2c_2^3 - c_2c_3)e_n^3 + (-12c_2^4 + 15c_2^2c_3 - 2c_3^2 - 2c_2c_4)e_n^4 \\ & + (43c_2^5 - 89c_2^3c_3 + 23c_2^2c_4 - 7c_3c_4 + c_2(33c_3^2 - 3c_5))e_n^5 + O(e_n^6). \end{aligned} \tag{3.6}$$

We get from (3.3) and (3.4)

$$\begin{aligned} s_n = \frac{f(z_n)}{f(y_n)} = {} & (2c_2^2 - c_3)e_n^2 - 2(3c_2^3 - 4c_2c_3 + c_4)e_n^3 \\ & + (9c_2^4 - 25c_2^2c_3 + 7c_3^2 + 11c_2c_4 - 3c_5)e_n^4 \\ & + 2\left(c_2^5 - 18c_2^3c_3 + 15c_2^2c_4 - 9c_3c_4 + c_2(16c_3^2 - 7c_5)\right)e_n^5 + O(e_n^6). \end{aligned} \tag{3.7}$$

Expanding J at $(0,0)$ and G at 0 yields

$$J(t_n, u_n) = J_{0,0} + u_n J_{0,1} + t_n J_{1,0} + \frac{1}{2}t_n^2 J_{2,0} + \frac{1}{6}t_n^3 J_{3,0} + O(t_n^4, u_n^2), \tag{3.8}$$

$$G(s_n) = G_0 + s_n G_1 + O(s_n^2). \tag{3.9}$$

Substituting (3.1)–(3.9) into (2.3), we obtain

$$e_{n+1} = x_{n+1} - x^* = R_4 e_n^4 + R_5 e_n^5 + R_6 e_n^6 + R_7 e_n^7 + R_8 e_n^8 + O(e_n^9),$$

where

$$\begin{aligned} R_4 &= -(2c_2^3 - c_2c_3)(-1 + G_0 J_{0,0}), \\ R_5 &= -c_2^2(2c_2^2 - c_3)(-2 + J_{1,0}), \\ R_6 &= -\frac{1}{2}c_2(2c_2^2 - c_3)\left(-2c_3(-1 + G_1) + c_2^2(-12 + 4G_1 + J_{2,0})\right), \\ R_7 &= -\frac{1}{6}c_2^2(2c_2^2 - c_3)\left(-6c_3(-2 + J_{0,1}) + c_2^2(-60 + 12J_{0,1} + J_{3,0})\right). \end{aligned}$$

By setting $R_4 = \cdots = R_7 = 0$ and $R_8 \neq 0$, the convergence order becomes eight. Obviously, we have

$$\begin{aligned} J_{0,0} = 1, \quad G_0 = 1 \quad &\Rightarrow \quad R_4 = 0, \\ J_{1,0} = 2, \quad &\Rightarrow \quad R_5 = 0, \\ J_{2,0} = 8, \quad G_1 = 1 \quad &\Rightarrow \quad R_6 = 0, \\ J_{0,1} = 2, \quad J_{3,0} = 36 \quad &\Rightarrow \quad R_7 = 0. \end{aligned}$$

Consequently, the error equation becomes in this case

$$e_{n+1} = \left(c_2 \left(2c_2^2 - c_3\right) \left(23c_2^4 - 12c_2^2c_3 + c_3^2 + c_2c_4\right)\right) e_n^8 + O(e_n^9)$$

which finishes the proof of the theorem. □

In what follows, we give some concrete explicit representations of (2.3) by choosing different weight functions satisfying the required conditions for the weight functions $J(t_n, u_n)$ and $G(s_n)$ of Theorem 1.

We can choose the weight functions $J(t_n, u_n)$ and $G(s_n)$ as

$$J(t_n, u_n) = \frac{1 + at_n + (2 + b)u_n + (2a + 1)t_n^2 + 4at_n^3}{1 + (a - 2)t_n + bu_n + t_n^2} \tag{3.10}$$

and

$$G(s_n) = \frac{1 + cs_n}{1 + (c - 1)s_n} \tag{3.11}$$

with arbitrary $a, b, c \in \mathbb{C}$. It is a simple task to check that the functions $J(t_n, u_n)$ and $G(s_n)$ in (3.10) and (3.11) satisfy the assumptions of Theorem 1 for all choices of a, b, c. Hence, three-parametric family of optimal eighth-order iterative root-finding methods is obtained.

By fixing the particular parameters a, b, c, we are going to give two examples of this family of methods.

Method 1: Set $a = b = c = \dfrac{1}{2}$. Then, we get

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(x_n)f(y_n)}{(f(x_n) - f(y_n))^2} \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] x_{n+1} := z_n - \dfrac{f(z_n)}{f'(x_n)} \left(\dfrac{2 + t_n + 5u_n + 4t_n^2 + 4t_n^3}{2 - 3t_n + u_n + 2t_n^2} \cdot \dfrac{2 + s_n}{2 - s_n} \right) \end{cases} \tag{3.12}$$

with $t_n = \frac{f(y_n)}{f(x_n)}$, $u_n = \frac{f(z_n)}{f(x_n)}$, $s_n = \frac{f(z_n)}{f(y_n)}$.

Method 2: Set $a = \dfrac{i + 1}{2}$, $b = 1 + i$, and $c = \dfrac{i - 1}{2}$. So, we have

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(x_n)f(y_n)}{(f(x_n) - f(y_n))^2} \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] x_{n+1} := z_n - \dfrac{f(z_n)}{f'(x_n)} \left(\dfrac{1 + (\frac{i+1}{2})t_n + (i + 3)u_n + (i + 2)t_n^2 + 4(\frac{i+1}{2})t_n^3}{1 + (\frac{i-3}{2})t_n + (i + 1)u_n + t_n^2} \cdot \dfrac{1 + (\frac{i-1}{2})s_n}{1 + (\frac{i-3}{2})s_n} \right) \end{cases} \tag{3.13}$$

with $t_n = \frac{f(y_n)}{f(x_n)}$, $u_n = \frac{f(z_n)}{f(x_n)}$, $s_n = \frac{f(z_n)}{f(y_n)}$.

We will apply in the next sections the new methods (3.12) and (3.13) to several benchmark examples and will compare the new methods with some existing optimal three-point methods of order eight having the same optimal computational efficiency index equal to $\sqrt[4]{8} \simeq 1.68179$, see [18, 24].

The existing methods that we are going to use to compare are the following:

Method 3: The method by Chun and Lee [5] is given by

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(y_n)}{f'(x_n)} \cdot \dfrac{1}{\left(1 - \frac{f(y_n)}{f(x_n)}\right)^2}, \\[2ex] x_{n+1} := z_n - \dfrac{f(z_n)}{f'(x_n)} \cdot \dfrac{1}{(1 - H(t_n) - J(s_n) - P(u_n))^2} \end{cases} \tag{3.14}$$

with weight functions

$$H(t_n) = -\beta - \gamma + t_n + \frac{t_n^2}{2} - \frac{t_n^3}{2}, \quad J(s_n) = \beta + \frac{s_n}{2}, \quad P(u_n) = \gamma + \frac{u_n}{2},$$

where $t_n = \frac{f(y_n)}{f(x_n)}$, $s_n = \frac{f(z_n)}{f(x_n)}$, $u_n = \frac{f(z_n)}{f(y_n)}$, and $\beta, \gamma \in \mathbb{R}$. Note that the parameters β and γ cancel when used in (3.14). Hence, their choice has no contribution to the method.

Method 4: The method by B. Neta [16], see also [17, formula (9)], is given by

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(x_n) + Af(y_n)}{f(x_n) + (A-2)f(y_n)} \cdot \dfrac{f(y_n)}{f'(x_n)}, \quad A \in \mathbb{R}, \\[2ex] x_{n+1} := y_n + \delta_1 f^2(x_n) + \delta_2 f^3(x_n), \end{cases} \tag{3.15}$$

where

$$\begin{aligned} F_y &= f(y_n) - f(x_n), & F_z &= f(z_n) - f(x_n), \\ \zeta_y &= \frac{1}{F_y}\left(\frac{y_n - x_n}{F_y} - \frac{1}{f'(x_n)}\right), & \zeta_z &= \frac{1}{F_z}\left(\frac{z_n - x_n}{F_z} - \frac{1}{f'(x_n)}\right), \\ \delta_2 &= -\frac{\zeta_y - \zeta_z}{F_y - F_z}, & \delta_1 &= \zeta_y + \delta_2 F_y. \end{aligned}$$

We will use $A = 0$ in the numerical experiments of this paper.

Method 5: The Sharma and Sharma method [22] is given by

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}, \\[2ex] z_n := y_n - \dfrac{f(y_n)}{f'(x_n)} \cdot \dfrac{f(x_n)}{f(x_n) - 2f(y_n)}, \\[2ex] x_{n+1} := z_n - \dfrac{f[x_n, y_n]f(z_n)}{f[x_n, z_n]f[y_n, z_n]} W(t_n), \end{cases} \tag{3.16}$$

with the weight function

$$W(t_n) = 1 + \frac{t_n}{1 + \alpha t_n}, \quad \alpha \in \mathbb{R},$$

and $t_n = \frac{f(z_n)}{f(x_n)}$. We will use $\alpha = 1$ in the numerical experiments of this paper.

Method 6: The method from Babajee, Cordero, Soleymani and Torregrosa [2] is given by

$$\begin{cases} y_n := x_n - \dfrac{f(x_n)}{f'(x_n)}\left(1+\left(\dfrac{f(x_n)}{f'(x_n)}\right)^5\right), \\[2ex] z_n := y_n - \dfrac{f(y_n)}{f'(x_n)}\left(1-\dfrac{f(y_n)}{f(x_n)}\right)^{-2}, \\[2ex] x_{n+1} := z_n - \dfrac{f(z_n)}{f'(x_n)} \cdot \dfrac{1+\left(\dfrac{f(y_n)}{f(x_n)}\right)^2+5\left(\dfrac{f(y_n)}{f(x_n)}\right)^4+\dfrac{f(z_n)}{f(y_n)}}{\left(1-\dfrac{f(y_n)}{f(x_n)}-\dfrac{f(z_n)}{f(x_n)}\right)^2}. \end{cases} \tag{3.17}$$

4 Numerical examples

The particular cases (3.12) and (3.13) of the the three-point method (2.3) are tested on a number of nonlinear equations. To obtain a high accuracy and avoid the loss of significant digits, we employed multi-precision arithmetic with 20 000 significant decimal digits in the programming package Mathematica.

In order to test our proposed methods (3.12) and (3.13), and also to compare them with the methods (3.14), (3.15), (3.16), and (3.17), we compute the error, the computational order of convergence (COC) by the approximate formula [27]

$$\mathrm{COC} \approx \frac{\ln |(x_{n+1}-x^*)/(x_n-x^*)|}{\ln |(x_n-x^*)/(x_{n-1}-x^*)|}, \tag{4.1}$$

and the approximated computational order of convergence (ACOC) by the formula [6]

$$\mathrm{ACOC} \approx \frac{\ln |(x_{n+1}-x_n)/(x_n-x_{n-1})|}{\ln |(x_n-x_{n-1})/(x_{n-1}-x_{n-2})|}. \tag{4.2}$$

It is worth noting that COC has been used in the recent years. Nevertheless, ACOC is more practical because it does not require to know the root x^*. For a comparison among several convergence orders, see [9]. Moreover, we should note that the results for these formulas not always coincide with or approximate the exact convergence order of the method when they are applied to a particular example. The reason is that we have in the error equations of the methods some coefficients that depend on c_k (see the proof of Theorem 1). Hence, these c_k's may vanish or vary for different kinds of examples. But, in general, a “random” example should provide good approximations for the order of convergence of the method.

On the other hand, it is nice to note that, given an iterative method, computing COC or ACOC on several examples is a good experiment to check theoretical errors in the deduction of the method and to check practical errors in the implementation of the method in a computer. For a general problem it will be difficult that COC or ACOC approach the theoretical order of convergence by chance.

We have used both COC and ACOC for checking the accuracy of the considered methods. Note that both COC and ACOC give already for small values of n good experimental approximations to convergence order.

In what follows, we are going to perform this kind of numerical experiments with the four test functions $f_j(x)$, $j = 1, \ldots, 4$, that appear in Table 1. In every case, and using the six eighth-order iterative methods described in the paper, we are going to reach the root x^* starting in the point x_0.

In Table 2, our new three-point methods (3.12) and (3.13) are tested on the four nonlinear equations $f_j(x) = 0$, $j = 1, 2, 3, 4$, and compared them with the methods (3.14), (3.15), (3.16), and (3.17) from other authors. We abbreviate (3.12)–(3.17) as M1–M6. Notice that, to estimate the COC and the

test function f_j	root x^*	initial guess x_0
$f_1(x) = \ln(1+x^2) + e^{x^2-3x}\sin x$	0	0.35
$f_2(x) = 1 + e^{2+x-x^2} + x^3 - \cos(1+x)$	-1	-0.3
$f_3(x) = (1+x^2)\cos\frac{\pi x}{2} + \frac{\ln(x^2+2x+2)}{1+x^2}$	-1	-1.1
$f_4(x) = x^4 + \sin\frac{\pi}{x^2} - 5$	$\sqrt{2}$	1.5

Table 1: Test functions $f_1, \ldots, f_4$, root x^*, and initial guess x_0.

ACOC, it has been enough to use $n = 3$ in (4.1) and (4.2) to get excellent approximations of the order of convergence.

	M1	M2	M3	M4	M5	M6
f_1, $x_0 = 0.35$						
$\lvert x_1 - x^*\rvert$	0.140e−3	0.318e−3	0.721e−4	0.893e−4	0.753e−4	0.347e−3
$\lvert x_2 - x^*\rvert$	0.583e−28	0.562e−25	0.230e−30	0.126e−30	0.619e−31	0.471e−25
$\lvert x_3 - x^*\rvert$	0.362e−223	0.531e−199	0.252e−242	0.200e−245	0.128e−247	0.546e−200
COC	8.0000	8.0000	8.0000	8.0000	8.0000	8.0000
ACOC	7.9999	8.0000	7.9999	7.9999	7.9999	7.9999
f_2, $x_0 = -0.3$						
$\lvert x_1 - x^*\rvert$	0.526e−4	0.113e−3	0.157e−3	0.763e−4	0.871e−4	0.411e−3
$\lvert x_2 - x^*\rvert$	0.534e−37	0.263e−33	0.119e−33	0.540e−35	0.134e−34	0.377e−29
$\lvert x_3 - x^*\rvert$	0.599e−301	0.226e−270	0.138e−274	0.342e−284	0.438e−281	0.189e−237
COC	8.0000	8.0000	8.0000	8.0000	8.0000	8.0000
ACOC	7.9999	7.9999	7.9998	7.9999	7.9999	7.9999
f_3, $x_0 = -1.1$						
$\lvert x_1 - x^*\rvert$	0.235e−7	0.298e−7	0.614e−8	0.388e−8	0.175e−8	0.554e−8
$\lvert x_2 - x^*\rvert$	0.393e−60	0.373e−59	0.328e−65	0.254e−67	0.154e−70	0.426e−66
$\lvert x_3 - x^*\rvert$	0.239e−482	0.222e−474	0.217e−523	0.877e−541	0.5821e−567	0.528e−531
COC	8.0000	8.0000	8.0000	8.0000	8.0000	8.0000
ACOC	7.9999	7.9999	8.0000	7.9999	8.0000	8.0000
f_4, $x_0 = 1.5$						
$\lvert x_1 - x^*\rvert$	0.286e−8	0.602e−8	0.433e−8	0.327e−10	0.642e−10	0.281e−8
$\lvert x_2 - x^*\rvert$	0.108e−68	0.181e−65	0.134e−66	0.369e−84	0.101e−81	0.341e−68
$\lvert x_3 - x^*\rvert$	0.460e−552	0.121e−525	0.116e−534	0.967e−676	0.389e−656	0.161e−547
COC	8.0000	8.0000	8.0000	8.0000	8.0000	8.0000
ACOC	8.0000	8.0000	7.9999	7.9999	7.9999	8.0000

Table 2: Errors, COC, and ACOC for the iterative methods (3.12)–(3.17) (abbreviated as M1–M6) applied to the find the root of test functions $f_1, \ldots, f_4$ given in Table 1.

5 Dynamic behavior

We already observed that all methods converge if the initial guess is chosen suitably. We now investigate the regions where we must choose the initial point to achieve the root. In other words, we numerically approximate the domain of attraction of the zeros as a qualitative measure of how demanding is the method on the initial approximation of the root. To answer the important question on the dynamical behavior of the algorithms, we investigate the dynamics of the new methods (3.12) and (3.13) and compare with common and well-performing methods from the literature, namely (3.14), (3.15), (3.16),

and (3.17). We recall in the following line some basic concepts such as basin of attraction. For more details and many other examples of the study of the dynamic behavior for iterative methods, one can consult [1, 2, 4, 7, 8, 10, 11, 23, 25].

Let $Q : \mathbb{C} \to \mathbb{C}$ be a rational map on the complex plane. For $z \in \mathbb{C}$, we define its orbit as the set $\mathrm{orb}(z) = \{z, Q(z), Q^2(z), \dots\}$. A point $z_0 \in \mathbb{C}$ is called periodic point with minimal period m if $Q^m(z_0) = z_0$ where m is the smallest positive integer with this property (and thus $\{z_0, Q(z_0), \dots, Q^{m-1}(z_0)\}$ is a cycle). A periodic point with minimal period 1 is called fixed point. Moreover, a fixed point z_0 is called attracting if $|Q'(z_0)| < 1$, repelling if $|Q'(z_0)| > 1$, and neutral otherwise. The Julia set of a nonlinear map $Q(z)$, denoted by $J(Q)$, is the closure of the set of its repelling periodic points. The complement of $J(Q)$ is the Fatou set $F(Q)$.

In our case, the six methods (3.12)–(3.17) provide iterative rational maps $Q(z)$ when they are applied to find the roots of complex polynomials $p(z)$. In particular, we are interesting in the basins of attraction of the roots of the polynomials where the basin of attraction of a root z^* is the complex set $\{z_0 \in \mathbb{C} : \mathrm{orb}(z_0) \to z^*\}$. It is well known that the basins of attraction of the different roots lie in the Fatou set $F(Q)$. The Julia set $J(Q)$ is, in general, a fractal and, in it, the rational map Q is unstable.

For the dynamical and graphical point of view, we take a 512×512 grid of the square $[-3, 3] \times [-3, 3] \subset \mathbb{C}$ and assign a color to each point $z_0 \in D$ according to the simple root to which the corresponding orbit of the iterative method starting from z_0 converges, and we mark the point as black if the orbit does not converge to a root in the sense that after at most 15 iterations it has a distance to any of the roots which is larger than 10^{-3}. We have used only 15 iterations because we are using eighth-order methods so, if the method converges, it is usually very fast. In this way, we distinguish the attraction basins by their color.

Test polynomials	Roots
$p_1(z) = z^2 - 1$	$1, \quad -1$
$p_2(z) = z^3 - z$	$0, \quad 1, \quad -1$
$p_3(z) = z(z^2 + 1)(z^2 + 4)$	$0, \quad 2i, \quad -2i, \quad i, \quad -i$
$p_4(z) = (z^4 - 1)(z^2 + 2i)$	$1, \quad i, \quad -1, \quad -i, \quad -1 + i, \quad 1 - i$
$p_5(z) = z^7 - 1$	$e^{2k\pi i/7}, \qquad k = 0, \dots, 6$
$p_6(z) = (10z^5 - 1)(z^5 + 10)$	$(\frac{1}{10})^{1/5} e^{2k\pi i/5}, \quad (-10)^{1/5} e^{2k\pi i/5}, \qquad k = 0, \dots, 4$

Table 3: Test polynomials $p_1(z), \dots, p_6(z)$ and their roots.

Different colors are used for different roots. In the basins of attraction, the number of iterations needed to achieve the root is shown by the brightness. Brighter color means less iteration steps. Note that black color denotes lack of convergence to any of the roots. This happens, in particular, when the method converges to a fixed point that is not a root or if it ends in a periodic cycle or at infinity. Actually and although we have not done it in this paper, infinity can be considered an ordinary point if we consider the Riemann sphere instead of the complex plane. In this case, we can assign a new "ordinary color" for the basin of attraction of infinity. Details for this idea can be found in [11].

Basins of attraction for the six methods (3.12)–(3.17) for the six test problems $p_i(z) = 0$, $i = 1, \dots, 6$, are illustrated in Figures 1–6 from left to right and from top to bottom.

From the pictures, we can easily judge the behavior and suitability of any method depending on the circumstances. If we choose an initial point z_0 in a zone where different basins of attraction touch each other, it is impossible to predict which root is going to be reached by the iterative method that starts in z_0. Hence, it is not a good choice. Both the black zones and the zones with a lot of colors are not suitable to take the initial guess z_0 when we want to achieve a precise root. The most attractive pictures appear when we have very intricate frontiers between basins of attraction and they correspond to the cases where the method is more demanding with respect to the initial point and its dynamic behavior is more unpredictable.

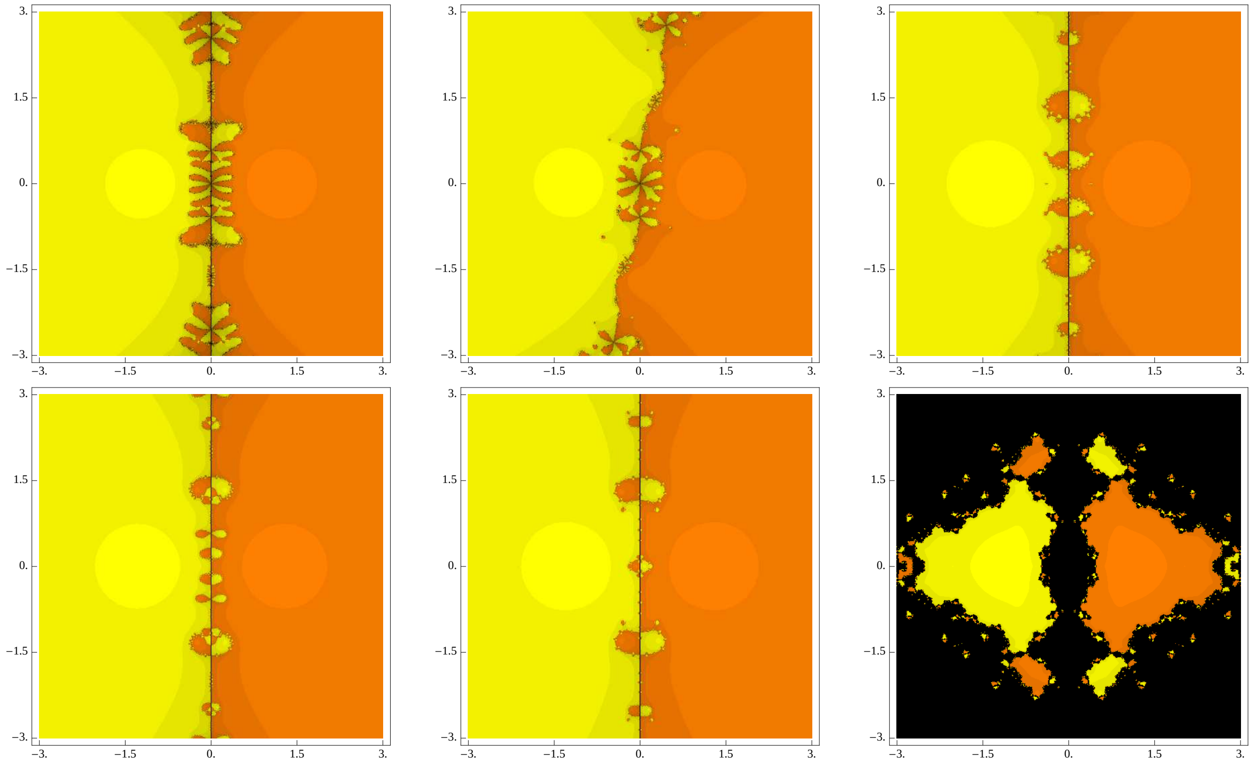

Figure 1: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_1(z) = z^2 - 1 = 0$.

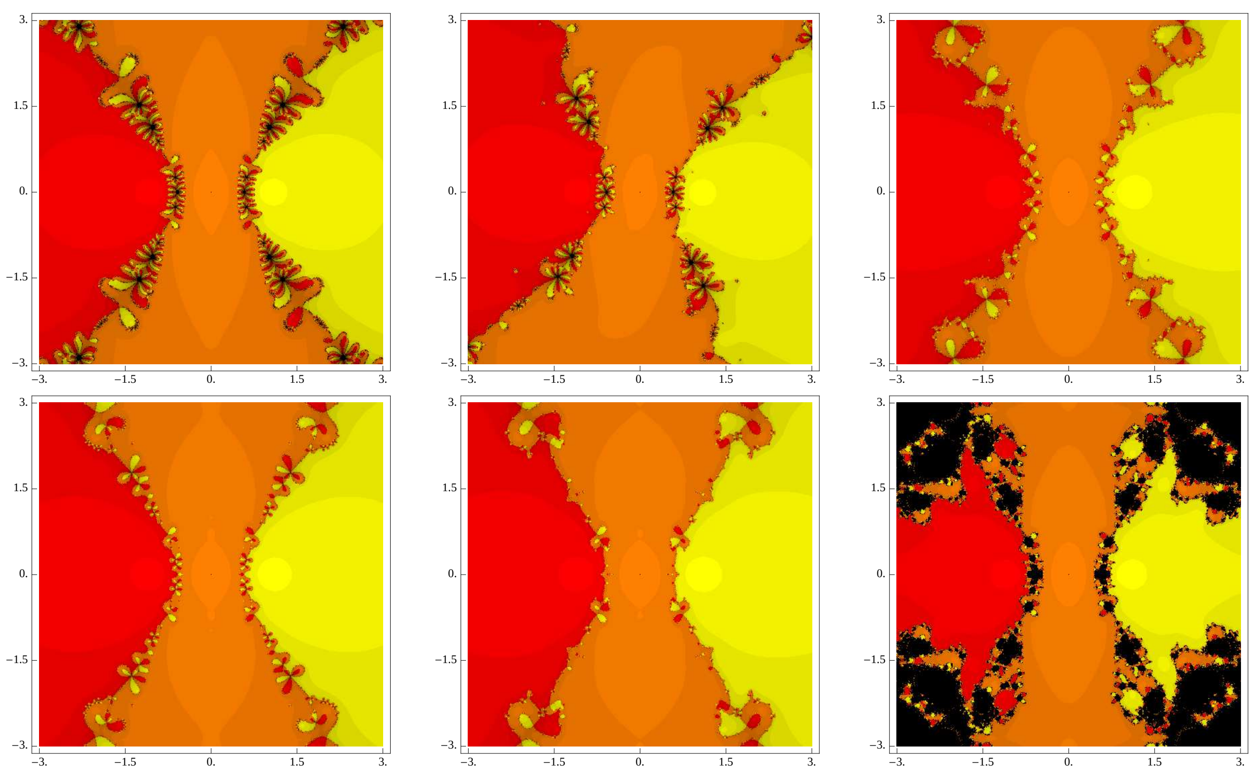

Figure 2: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_2(z) = z^3 - z = 0$.

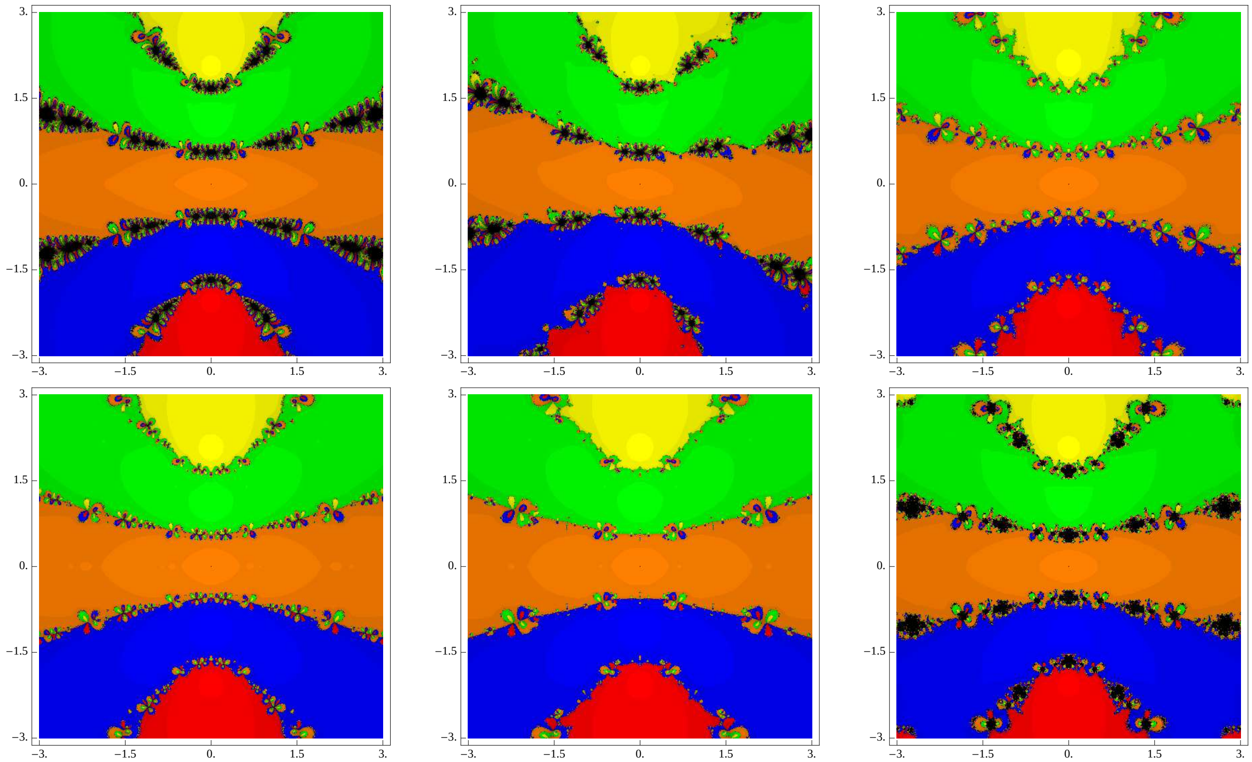

Figure 3: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_3(z) = z(z^2+1)(z^2+4) = 0$.

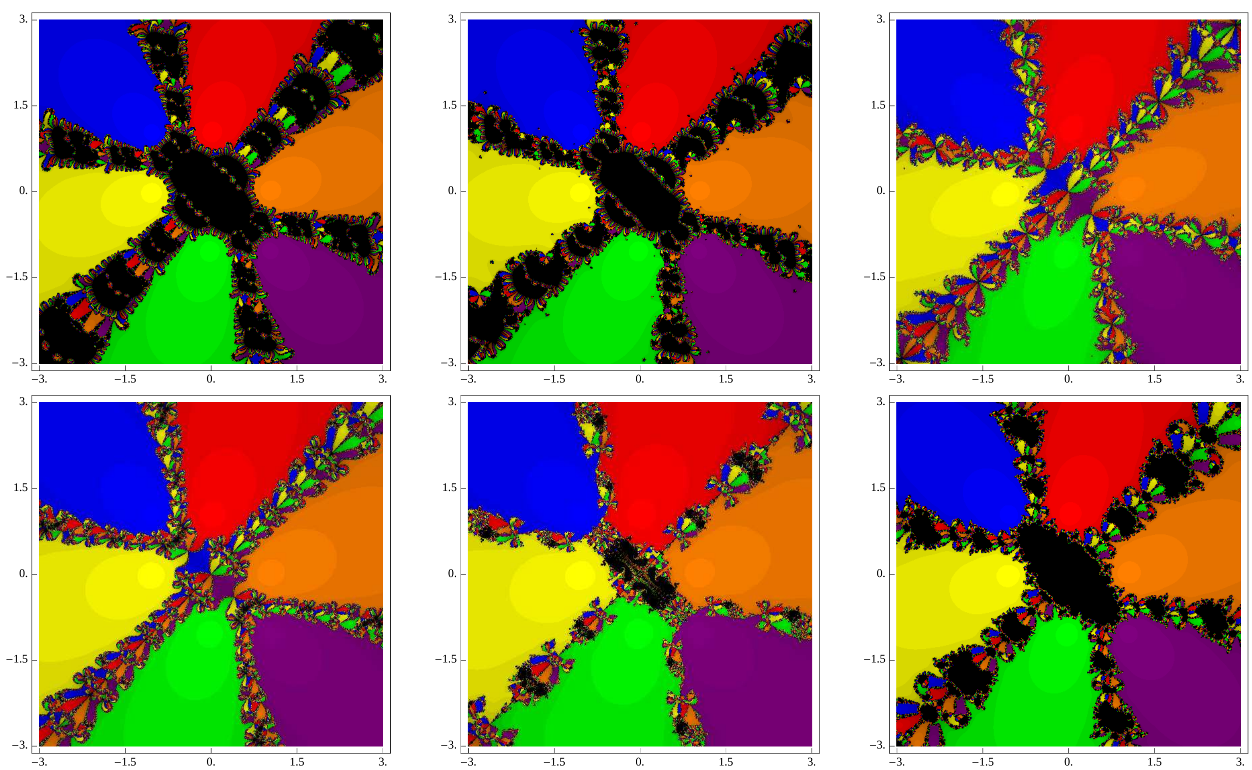

Figure 4: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_4(z) = (z^4-1)(z^2+2i) = 0$.

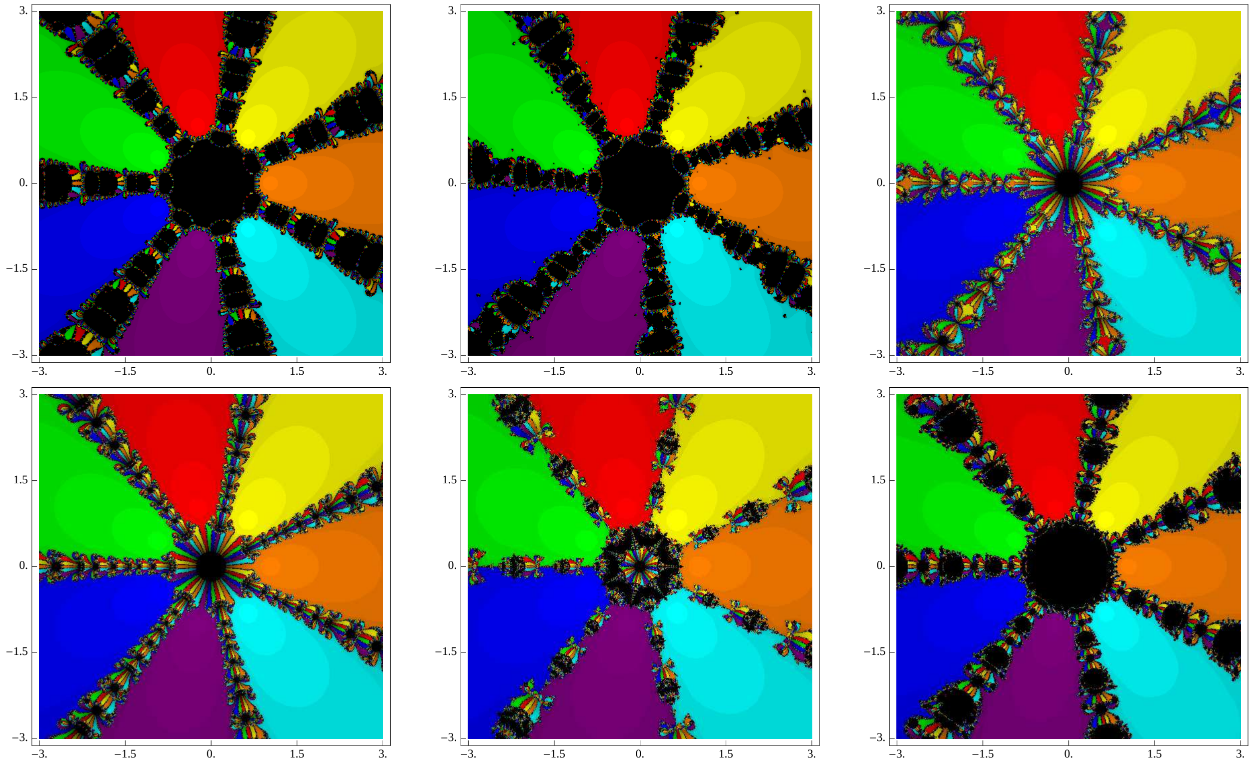

Figure 5: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_5(z) = z^7 - 1 = 0$.

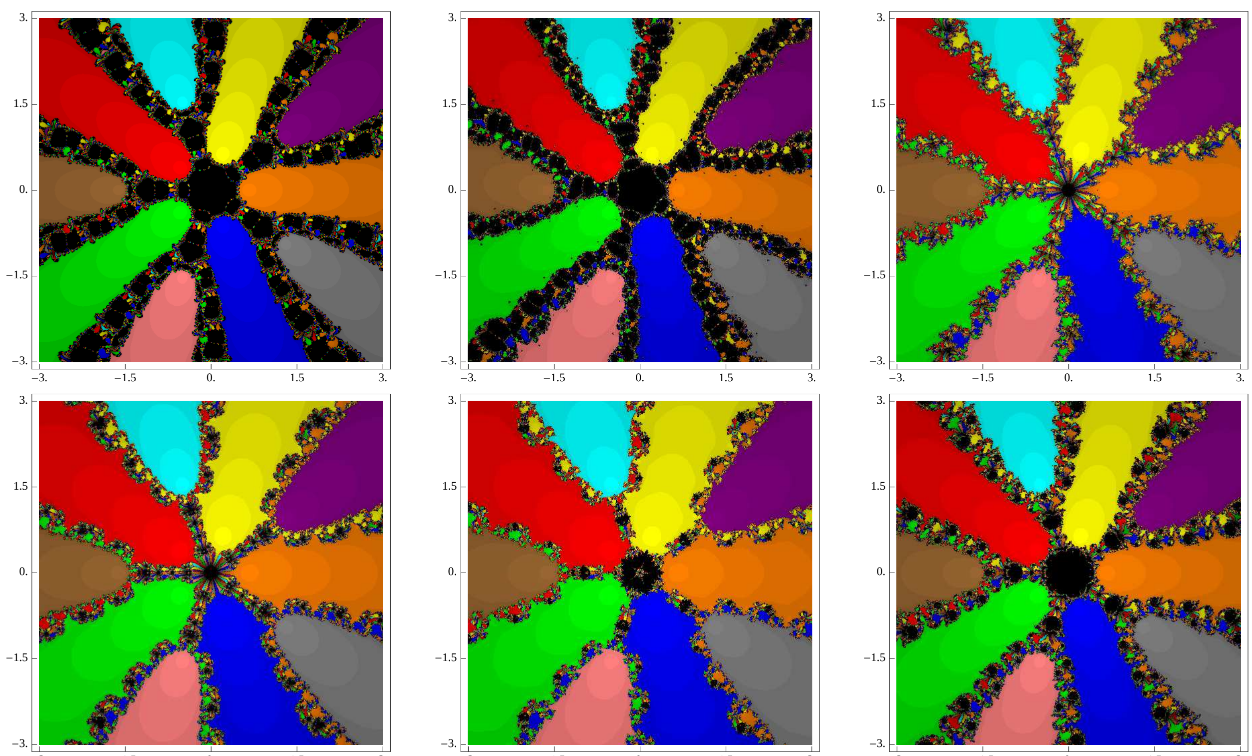

Figure 6: Comparison of basins of attraction of methods (3.12)–(3.17) for the test problem $p_6(z) = (10z^5 - 1)(z^5 + 10) = 0$.

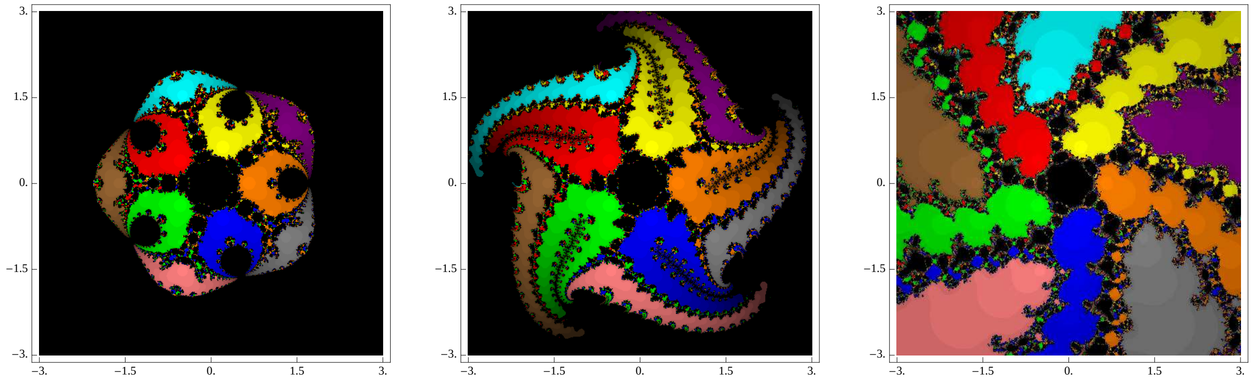

Figure 7: Basins of attraction of the method of Theorem 1 to solve $p_6(z) = (10z^5 - 1)(z^5 + 10) = 0$ with different choices of the parameters a, b and c. Left: $a = b = c = -1$. Center: $a = b = -1$, $c = -1 + i$. Right: $a = -1/2$, $b = -1$, $c = -2 + i$.

The first and the second graphics in the Figures 1–6 correspond to the same general method (2.3) with different choices of the parameters a, b, c in (3.10) and (3.11). We can see from the graphics in Figure 7 and the first two pictures of Figure 6 that even small changes in the parameters may lead to completely different behaviors.

Finally, we have included in Table 4 the results of some numerical experiments to measure the behavior of the six iterative methods (3.12)–(3.17) in finding the roots of the test polynomials $p_j(z)$, $j = 1, \dots, 6$. To compute the data of this table, we have applied the six methods to the six polynomials, starting at an initial points z_0 on a 512×512 grid in the rectangle $[-3, 3] \times [-3, 3]$ of the complex plane. The same way was used in Figures 1–6 to show the basins of attraction of the roots. In particular, we decide again that an initial point z_0 has reached a root z^* when its distance to z^* is less than 10^{-3} (in this case z_0 is in the basin of attraction of z^*) and we decide that the method starting in z_0 diverges when no root is found in a maximum of 15 iterations of the method (in this case, we say that z_0 is a "nonconvergent point"). In Table 4, we have abbreviated the methods (3.12)–(3.17) as M1–M6, respectively. The column I/P shows the mean of iterations per point until the algorithm decides that a root has been reached or the point is declared nonconvergent. The column NC shows the percentage of nonconvergent points, indicated as black zones in the figures. It is clear that the nonconvergent points have a great influence on the values of I/P since these points contribute always with the maximum number of 15 allowed iterations, In contrast, "convergent points" are reached usually very fast due to the fact that we are dealing with eighth-order methods. To reduce the effect of nonconvergent points, we have included the column $\mathrm{I_C/C}$ which shows the mean number of iterations per convergent point. If we use either the columns I/P or the column $\mathrm{I_C/C}$ to compare the performance of the iterative methods, we clearly obtain different conclusions.

6 Conclusion

We have introduced a new optimal class of three-point methods without memory for approximating a simple root of a given nonlinear equation which use only four function evaluations each iteration and result in a method of convergence order eight. Therefore, the Kung and Traub's conjecture is supported. Numerical examples and comparisons with some existing eighth-order methods are included and confirm the theoretical results. The numerical experience suggests that the new class is a valuable alternative for solving these problems and finding simple roots. We used the basins of attraction for comparing the iterative algorithms and we have included some tables with comparative results.

Polynomial	Method	I/P	NC (%)	I_C/C	Polynomial	Method	I/P	NC (%)	I_C/C
$p_1(z)$	M1	2.53	0.244	2.50	$p_4(z)$	M1	6.85	24.7	4.17
	M2	2.29	0.00798	2.28		M2	6.48	22.0	4.07
	M3	2.20	0.195	2.18		M3	4.07	0.888	3.97
	M4	2.17	0.195	2.15		M4	4.21	1.84	4.01
	M5	2.13	0.195	2.10		M5	3.95	4.40	3.44
	M6	6.01	70.9	2.09		M6	4.45	20.1	3.56
$p_2(z)$	M1	3.54	0.798	3.45	$p_5(z)$	M1	7.27	27.0	4.42
	M2	3.10	0.340	3.06		M2	7.00	25.2	4.30
	M3	2.88	0.	2.88		M3	4.81	3.36	4.45
	M4	2.82	0.00456	2.82		M4	5.07	5.71	4.47
	M5	2.73	0.	2.73		M5	4.59	7.04	3.80
	M6	4.32	27.6	2.81		M6	5.03	21.4	4.02
$p_3(z)$	M1	3.88	3.57	3.47	$p_6(z)$	M1	7.36	24.4	4.90
	M2	3.57	2.19	3.31		M2	6.96	21.7	4.73
	M3	2.99	0.0122	2.99		M3	4.69	2.33	4.44
	M4	2.94	0.0334	2.94		M4	4.89	4.03	4.46
	M5	2.82	0.	2.82		M5	4.44	3.98	4.01
	M6	3.28	5.46	2.99		M6	5.26	11.9	4.70

Table 4: Measures of convergence of the iterative methods (3.12)–(3.17) (abbreviated as M1–M6) applied to find the roots of the polynomials $p_j(z)$, $j = 1, \dots, 6$.

References

[1] Amat, S., Busquier, S., Magreñán, Á.A.: Reducing chaos and bifurcations in Newton-type methods, Abstr. Appl. Anal. 2013, Art. ID 726701, 10 pages (2013).

[2] Babajee, D.K.R., Cordero, A., Soleymani, F., Torregrosa, J.R.: On improved three-step schemes with high efficiency index and their dynamics, Numer. Algorithms 65, 153–169 (2014).

[3] Bi, W., Ren, H., Wu, Q.: Three-step iterative methods with eighth-order convergence for solving nonlinear equations, J. Comput. Appl. Math. 225, 105–112 (2009).

[4] Chicharro, F., Cordero, A., Gutiérrez, J.M., Torregrosa, J.R.: Complex dynamics of derivative-free methods for nonlinear equations, Appl. Math. Comput. 219, 7023–7035 (2013).

[5] Chun, C., Lee, M.Y.: A new optimal eighth-order family of iterative methods for the solution of nonlinear equations, Appl. Math. Comput. 223, 506–519 (2013).

[6] Cordero, A., Torregrosa, J.R.: Variants of Newton's method using fifth-order quadrature formulas, Appl. Math. Comput. 190, 686–698 (2007).

[7] Ezquerro, J.A., Hernández, M.A.: An improvement of the region of accessibility of Chebyshev's method from Newton's method, Math. Comp. 78, 1613–1627 (2009).

[8] Ezquerro, J.A., Hernández, M.A.: An optimization of Chebyshev's method, J. Complexity 25, 343–361 (2009).

[9] Grau-Sánchez, M., Noguera, M., Gutiérrez, J.M.: On some computational orders of convergence, Appl. Math. Lett. 23(4), 472–478 (2010).

[10] Gutiérrez, J.M., Magreñán, Á.A., Varona, J.L.: The "Gauss-Seidelization" of iterative methods for solving nonlinear equations in the complex plane, Appl. Math. Comput. 218, 2467–2479 (2011).

[11] Hernández-Paricio, L.J., Marañón-Grandes, M., Rivas-Rodríguez, M.T.: Plotting basins of end points of rational maps with Sage, Tbil. Math. J. 5(2), 71–99 (2012).

[12] Jarratt, P.: Some fourth order multipoint iterative methods for solving equations, Math. Comp. 20, 434–437 (1966).

[13] King, R.F.: A family of fourth order methods for nonlinear equations, SIAM J. Numer. Anal. 10, 876–879 (1973).

[14] Kung, H.T., Traub, J.F.: Optimal order of one-point and multipoint iteration, J. Assoc. Comput. Mach. 21, 634–651 (1974).

[15] Lotfi, T., Sharifi, S., Salimi, M., Siegmund, S.: A new class of three-point methods with optimal convergence order eight and its dynamics, Numer. Algorithms 68, 261–288 (2015).

[16] Neta, B.: On a family of multipoint methods for nonlinear equations, Internat. J. Comput. Math. 9, 353–361 (1981).

[17] Neta, B., Chun, C., Scott, M.: Basins of attraction for optimal eighth order methods to find simple roots of nonlinear equations, Appl. Math. Comput. 227, 567–592 (2014).

[18] Ostrowski, A.M.: Solution of Equations and Systems of Equations, 2nd ed., Academic Press, New York (1966).

[19] Petković, M.S., Neta, B., Petković, L.D., Džunić, J.: Multipoint Methods for Solving Nonlinear Equations, Elsevier/Academic Press, Amsterdam (2013).

[20] Sharifi, S., Ferrara, M., Salimi, M., Siegmund, S.: New modification of Maheshwari method with optimal eighth order of convergence for solving nonlinear equations, preprint (2015).

[21] Sharifi, S., Siegmund, S., Salimi, M.: Solving nonlinear equations by a derivative-free form of the King's family with memory, Calcolo, doi: 10.1007/s10092-015-0144-1 (2015).

[22] Sharma, J.R., Sharma, R.: A new family of modified Ostrowski's methods with accelerated eighth order convergence, Numer. Algorithms 54, 445–458 (2010).

[23] Stewart, B.D.: Attractor Basins of Various Root-Finding Methods, M.S. thesis, Naval Postgraduate School, Monterey, CA (2001).

[24] Traub, J.F.: Iterative Methods for the Solution of Equations, Prentice Hall, Englewood Cliffs, N.J. (1964).

[25] Varona, J.L.: Graphic and numerical comparison between iterative methods, Math. Intelligencer 24(1), 37–46 (2002).

[26] Wang, X., Liu, L.: New eighth-order iterative methods for solving nonlinear equations, J. Comput. Appl. Math. 234, 1611–1620 (2010).

[27] Weerakoon, S., Fernando, T.G.I.: A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 13(8), 87–93 (2000).